

UNERSIDAD PRIVADA DE TACNA



INGENIERIA DE SISTEMAS

TITULO:

INFORME DE LABORATORIO No 01

CURSO:

BASE DE DATOS II

DOCENTE:

Ing. Patrick Cuadros Quiroga

Integrantes:

Espinoza Caso, Lisbeth

(2011040667)

... Continuen poniendo sus nombres

Índice

1. Actividad No 01 – Revisión de Sintaxis	1
2. Actividad No 02 – Reconociendo la estructura	3
3. Actividad No 03 – Consultas Básicas	5
4. Actividad No 04 – Restricción y Ordenamiento	7
5. Actividad No 05 – Funciones	15
6. Actividad No 06 – Funciones de Conversión	19

1. Actividad No 01 – Revisión de Sintaxis

De los siguientes comandos ¿Cuál es el resultado? ¿En caso de ser error cual sería la sentencia correcta?

- SELECT last_name, job_id, salary AS Sal FROM employees;
Es correcta

```
SELECT last_name, job_id, salary AS Sal FROM employees;
```

	last_name	job_id	Sal
1	King	AD_PRES	24000.00
2	Kochhar	AD_VP	17000.00
3	De Haan	AD_VP	17000.00
4	Hunold	IT_PROG	9000.00
5	Ernst	IT_PROG	6000.00
6	Austin	IT_PROG	4800.00
7	Pataballa	IT_PROG	4800.00
8	Lorentz	IT_PROG	4200.00
9	Greenberg	FI_MGR	12008.00
10	Faviet	FI_ACC...	9000.00
11	Chen	FI_ACC...	8200.00
12	Sciarra	FI_ACC...	7700.00
13	Urman	FI_ACC...	7800.00
14	Popp	FI_ACC...	6900.00
15	Raphaely	PU_MAN	11000.00

- SELECT * FROM job_grades;
Es incorrecta, la sentencia correcta sería:
SELECT * FROM jobs;

```
SELECT * FROM jobs;
```

	job_id	job_title	min_salary	max_salary
1	AC_ACCOUNT	Public Accountant	4200	9000
2	AC_MGR	Accounting Manager	8200	16000
3	AD_ASST	Administration Assistant	3000	6000
4	AD_PRES	President	20080	40000
5	AD_VP	Administration Vice President	15000	30000
6	FI_ACCOUNT	Accountant	4200	9000
7	FI_MGR	Finance Manager	8200	16000
8	HR_REP	Human Resources Representative	4000	9000
9	IT_PROG	Programmer	4000	10000
10	MAK_MGR	Marketing Manager	9000	15000
11	MAK_REP	Marketing Representative	4000	9000
12	PR_REP	Public Relations Representative	4500	10500
13	PU_CLERK	Purchasing Clerk	2500	5500
14	PU_MAN	Purchasing Manager	8000	15000
15	SA_MAN	Sales Manager	10000	20000
16	SA_REP	Sales Representative	6000	12000
17	SH_CLERK	Shipping Clerk	2500	5500
18	ST_CLERK	Stock Clerk	2008	5000
19	ST_MAN	Stock Manager	5500	8500

- SELECT employee_id, last_name sal x 12 ANNUAL SALARY FROM employees;
Es incorrecta, la sentencia correcta sería:
SELECT employee_id, last_name, salary * 12 'ANNUAL SALARY' FROM employees;

```
~ SELECT employee_id, last_name, salary*12 'ANNUAL SALARY' FROM employees;  
RQ
```

Results		Messages	
	employee_id	last_name	ANNUAL SALARY
1	100	King	288000.00
2	101	Kochhar	204000.00
3	102	De Haan	204000.00
4	103	Hunold	108000.00
5	104	Emst	72000.00
6	105	Austin	57600.00
7	106	Pataballa	57600.00
8	107	Lorentz	50400.00
9	108	Greenberg	144096.00
10	109	Faviet	108000.00
11	110	Chen	98400.00
12	111	Sciarra	92400.00
13	112	Uman	93600.00
14	113	Popp	82800.00
15	114	Raphaely	132000.00

2. Actividad No 02 – Reconociendo la estructura

1. Se requiere determinar la estructura de la tabla DEPARTMENTS y sus datos.

SP_HELP 'DEPARTMENTS'

SP_HELP departments;

go

	Name	Owner	Type	Created_datetime
1	departments	dbo	user table	2018-09-28 21:48:06.947

	Column_name	Type	Computed	Length	Prec	Scale	Nullable	TrimTrailingBlanks	FixedLenNullInSource	Collation
1	department_id	int	no	4	10	0	no	(n/a)	(n/a)	NULL
2	department_name	varchar	no	30			yes	no	yes	SQL_Latin1_General_CP1_CI_AS
3	manager_id	int	no	4	10	0	yes	(n/a)	(n/a)	NULL
4	location_id	int	no	4	10	0	yes	(n/a)	(n/a)	NULL

	Identity	Seed	Increment	Not For Replication
1	No identity column defined.	NULL	NULL	NULL

	RowGuidCol
1	No rowguidcol column defined.

	Data_located_on_filegroup
1	PRIMARY

	index_name	index_description	index_keys
1	dept_id_pk	clustered, unique, primary key located on PRIMARY	department_id

	constraint_type	constraint_name	delete_action	update_action	status_enabled	status_for_replication	constraint_keys
1	PRIMARY KEY (clustered)	dept_id_pk	(n/a)	(n/a)	(n/a)	(n/a)	department_id
2	FOREIGN KEY	dept_loc_fk	No Action	No Action	Enabled	Is_For_Replication	location_id
3							REFERENCES Human_Resources.dbo.locations (locati...

	Table is referenced by foreign key
1	Human_Resources.dbo.employees: emp_dept_fk
2	Human_Resources.dbo.job_history: jhist_dept_fk

2. El departamento de Recursos Humanos requiere un reporte que muestre los campos: employee_id, last_name y job_id, asicomo el campo hire_date con el alias StartDate.

```
SELECT emp.employee_id,  
emp.last_name,  
emp.job_id,  
emp.hire_date AS StartDate  
FROM employees AS emp;
```

SELECT emp.employee_id,
emp.last_name,
emp.job_id,
emp.hire_date AS StartDate
FROM employees AS emp;

go

	employee_id	last_name	job_id	StartDate
1	100	King	AD_PRES	2003-06-17
2	101	Kochhar	AD_VP	2005-09-21
3	102	De Haan	AD_VP	2001-01-13
4	103	Hunold	IT_PROG	2006-01-03
5	104	Ernst	IT_PROG	2007-05-21
6	105	Austin	IT_PROG	2005-06-25
7	106	Pataballa	IT_PROG	2006-02-05
8	107	Lorentz	IT_PROG	2007-02-07
9	108	Greenberg	FI_MGR	2002-08-17
10	109	Faviet	FI_ACCOUNT	2002-08-16
11	110	Chen	FI_ACCOUNT	2005-09-28
12	111	Sciarra	FI_ACCOUNT	2005-09-30
13	112	Uman	FI_ACCOUNT	2006-03-07
14	113	Popp	FI_ACCOUNT	2007-12-07
15	114	Raphaely	PU_MAN	2002-12-07

3. Finalmente el departamento de Recursos Humanos requiere un listado de todos valores del campo JOB.ID de la tabla EMPLOYEES pero que se muestren de forma única y no repetida.

```
SELECT DISTINCT job_id FROM employees;
```

```
SELECT DISTINCT job_id FROM employees;  
go
```

Results		Messages	
	job_id		
1	AC_ACCOUNT		
2	AC_MGR		
3	AD_ASST		
4	AD PRES		
5	AD_VP		
6	FI_ACCOUNT		
7	FI_MGR		
8	HR_REP		
9	IT_PROG		
10	MK_MAN		
11	MK_REP		
12	PR_REP		
13	PU_CLERK		
14	PU_MAN		
15	SA_MAN		

3. Actividad No 03 – Consultas Básicas

1. El departamento de Recursos Humanos requiere ampliar el reporte anterior (4.2.2) para hacerlo más comprensible, por lo que se requiere que los encabezados de las columnas sean: Emp No, Empleado, Puesto y Fecha Contratación.

```
SELECT emp.employee_id AS 'Emp N',  
emp.last_name AS Empleado,  
emp.job_id AS Puesto,  
emp.hire_date AS 'Fecha de contratación'  
FROM employees AS emp;
```

```
:SELECT emp.employee_id AS 'Emp N',  
emp.last_name AS employee,  
emp.job_id AS jobs,  
emp.hire_date AS 'Fecha de contratación'  
FROM employees AS emp;  
go
```

	Emp N	employee	jobs	Fecha de contratación
1	100	King	AD_PRES	2003-06-17
2	101	Kochhar	AD_VP	2005-09-21
3	102	De Haan	AD_VP	2001-01-13
4	103	Hunold	IT_PROG	2006-01-03
5	104	Ernst	IT_PROG	2007-05-21
6	105	Austin	IT_PROG	2005-06-25
7	106	Pataballa	IT_PROG	2006-02-05
8	107	Lorentz	IT_PROG	2007-02-07
9	108	Greenberg	FI_MGR	2002-08-17
10	109	Faviet	FI_ACCOUNT	2002-08-16
11	110	Chen	FI_ACCOUNT	2005-09-28
12	111	Sciarra	FI_ACCOUNT	2005-09-30
13	112	Uman	FI_ACCOUNT	2006-03-07
14	113	Popp	FI_ACCOUNT	2007-12-07
15	114	Raphaely	PU_MAN	2002-12-07

2. Adicionalmente el departamento de Recursos Humanos requiere un reporte más sencillo, en el que se muestre los campos: last_name y job_id en una sola y única columna (los datos deben estar separados por una coma) que tenga como alias Empleado y Puesto.

```
SELECT CONCAT(emp.last_name,',',emp.job_id) AS 'Empleado y Puesto'  
  
FROM employees AS emp;
```

Empleado y Puesto	
1	King,AD_PRES
2	Kochhar,AD_VP
3	De Haan,AD_VP
4	Hunold,IT_PROG
5	Emst,IT_PROG
6	Austin,IT_PROG
7	Pataballa,IT_PROG
8	Lorentz,IT_PROG
9	Greenberg,FI_MGR
10	Faviet,FI_ACCOUNT
11	Chen,FI_ACCOUNT
12	Sciarra,FI_ACCOUNT
13	Uman,FI_ACCOUNT
14	Popp,FI_ACCOUNT
15	Raphaely,PU_MAN

```
SELECT CONCAT(emp.last_name,',',emp.job_id) AS 'Empleado y Puesto'
FROM employees AS emp;
go
```

3. Finalmente a modo de práctica, realizar una consulta que muestre todos los campos de la tabla EMPLOYEES, en una sola y única columna, los datos deben estar separados por una coma y la columna debe tener como encabezado Los Empleados

```
SELECT CONCAT(emp.employee_id,',',
emp.first_name,',',
emp.last_name,',',
emp.email,',',
emp.phone_number,',',
emp.hire_date,',',
emp.job_id,',',
emp.salary,',',
emp.commission_pct,',',
emp.manager_id,',',
emp.department_id) AS 'Los empleados'
FROM employees AS emp;
```

```
SELECT CONCAT(emp.employee_id,',',
emp.first_name,',',
emp.last_name,',',
emp.email,',',
emp.phone_number,',',
emp.hire_date,',',
emp.job_id,',',
emp.salary,',',
emp.commission_pct,',',
emp.manager_id,',',
emp.department_id) AS 'Los empleados'
FROM employees AS emp;
go
```

Los empleados	
1	100.Steven.King,SKING,515.123.4567,2003-06-17,AD_P...
2	101.Neena.Kochhar,NKOCHHAR,515.123.4568,2005-09-...
3	102.Lex.De Haan,LDEHAAN,515.123.4569,2001-01-13,A...
4	103.Alexander.Hunold,AHUNOLD,590.423.4567,2006-01-...
5	104.Bruce.Ernst,BERNST,590.423.4568,2007-05-21,IT_P...
6	105.David.Austin,DAUSTIN,590.423.4569,2005-06-25,IT_...
7	106.Valli.Pataballa,VPATABAL,590.423.4560,2006-02-05,I...
8	107.Diana.Lorentz,DLORENTZ,590.423.5567,2007-02-07,...
9	108.Nancy.Greenberg,NGREENBE,515.124.4569,2002-08-...
10	109.Daniel.Faviet,DFAVIET,515.124.4169,2002-08-16,FI_...
11	110.John.Chen,JCHEN,515.124.4269,2005-09-28,FI_ACC...
12	111.Ismail.Sciarra,ISCIARRA,515.124.4369,2005-09-30,FI...
13	112.Jose Manuel.Uman,JMURMAN,515.124.4469,2006-0...
14	113.Luis.Popp,LPOPP,515.124.4567,2007-12-07,FI_ACCO...
15	114.Den.Raphaely,DRAPHEAL,515.127.4561,2002-12-07,...

4. Actividad No 04 – Restricción y Ordenamiento

1. Debido a problemas con el presupuesto, el departamento de Recursos Humanos requiere un reporte que muestre los apellidos (last_name) y salarios (salary) de todos los empleados que ganen más de \$ 12,000.

```
select last_name,salary from employees where salary >12000;
```

	last_name	salary
1	King	24000.00
2	Kochhar	17000.00
3	De Haan	17000.00
4	Greenberg	12008.00
5	Russell	14000.00
6	Partners	13500.00
7	Hartstein	13000.00
8	Higgins	12008.00

2. Asimismo se requiere realizar una consulta que muestre los apellidos (last_name) y el número de departamento (department_id) para los empleados que tengan numero (employee_id) 176.

```
select last_name,department_id from employees where employee_id <176;
```

	last_name	department_id
1	Livingston	80
2	Grant	NULL
3	Johnson	80
4	Taylor	50
5	Fleur	50
6	Sullivan	50
7	Geoni	50
8	Sarchand	50
9	Bull	50
10	Dellinger	50
11	Cabrio	50
12	Chung	50
13	Dilly	50
14	Gates	50
15	Perkins	50
16	Bell	50
17	Everett	50
18	McCain	50
19	Jones	50

3. El departamento de Recursos Humanos necesita determinar los mayores y menores sueldos, modificar la consulta del ítem 4.1. para mostrar el apellido y salario de cada empleado cuyo sueldo no esté en el rango de \$ 5,000 a \$ 12,000.

```
select last_name,job_id,salary as Sal from employees where salary <5000 and salary >12000;
```

	last_name	job_id	Sal
1	Hunold	IT_PROG	9000.00
2	Ernst	IT_PROG	6000.00
3	Faviet	FI_ACCOUNT	9000.00
4	Chen	FI_ACCOUNT	8200.00
5	Sciarra	FI_ACCOUNT	7700.00
6	Uman	FI_ACCOUNT	7800.00
7	Popp	FI_ACCOUNT	6900.00
8	Raphaely	PU_MAN	11000.00
9	Weiss	ST_MAN	8000.00
10	Frapp	ST_MAN	8200.00
11	Kaufling	ST_MAN	7900.00
12	Vollman	ST_MAN	6500.00
13	Mourgos	ST_MAN	5800.00
14	Cambraut	SA_MAN	11000.00
15	Zlotkey	SA_MAN	10500.00
16	Tucker	SA_REP	10000.00
17	Bernstein	SA_REP	9500.00
18	Hall	SA_REP	9000.00
19	Olsen	SA_REP	8000.00

4. Crear un reporte que muestre los apellidos (last_name), puesto (job_id) y fecha de contratación (hire_date), de los empleados que apellidan 'Matos' y 'Taylor', asimismo presentar el reporte ordenado ascendentemente por fecha de contratación.

```
select last_name,job_id,hire_date from employees where last_name = 'Matos' or last_name = 'Taylor' order by hire_date asc;
```

	last_name	job_id	hire_date
1	Taylor	SH_CLERK	2006-01-24
2	Matos	ST_CLERK	2006-03-15
3	Taylor	SA_REP	2006-03-24

5. Mostrar los apellidos (last_name) y número de departamento (departamento_id) de todos los empleados que pertenezcan a los departamentos 20 o 50 en orden alfabético ascendente por el apellido.

```
select last_name,department_id from employees where department_id = 20 or department_id = 50 order by last_name asc;
```

	last_name	department_id
1	Atkinson	50
2	Bell	50
3	Bissot	50
4	Bull	50
5	Cabrio	50
6	Chung	50
7	Davies	50
8	Dellinger	50
9	Dilly	50
10	Everett	50
11	Fay	20
12	Feeney	50
13	Fleaur	50
14	Fripp	50
15	Gates	50
16	Gee	50
17	Geoni	50
18	Grant	50
19	Hartstein	20

6. Modificar el reporte del ítem 4.1. para mostrar los apellidos y salarios de los empleados que tengan un salario entre los \$ 5,000 a \$ 12,000 y pertenezcan a los números de departamento 20 o 50. Asimismo etiquetar las cabeceras de los resultados con los alias Empleado y Salario Mensual respectivamente.

```
select last_name 'Empleado',salary 'Salario Mensual' from employees where salary >5000 and salary <12000 and (department_id = 20 or department_id = 50);
```

	Empleado	Salario Mensual
1	Weiss	8000.00
2	Fripp	8200.00
3	Kaufling	7900.00
4	Vollman	6500.00
5	Mourgos	5800.00
6	Fay	6000.00

7. El departamento de Recursos Humanos necesita un listado de apellidos (last_name) y fecha de contratación (hire_date) de todos los empleados que fueron contratados el año 1994.

```
select last_name,hire_date from employees where hire_date between '19940101' and '19941231';
```

last_name	hire_date
-----------	-----------

8. Crear un reporte que muestre los apellidos (last_name) y puesto (job_id) de todos los empleados que no tengan un administrador (manager).

```
select last_name,job_id from employees where manager_id is null;
```

	last_name	job_id
1	King	AD_PRES

9. Crear un reporte para mostrar los apellidos (last_name), salario (salary) y % de comisión (commission_pct). Ordenar los datos por salario y comisión de manera descendente, utilizar la opción numérica de la cláusula ORDER BY.

```
select last_name,salary,commission_pct from employees order by salary desc,commission_pct desc;
```

	last_name	salary	commission_pct
1	King	24000.00	NULL
2	Kochhar	17000.00	NULL
3	De Haan	17000.00	NULL
4	Russell	14000.00	0.40
5	Partners	13500.00	0.30
6	Hartstein	13000.00	NULL
7	Higgins	12008.00	NULL
8	Greenberg	12008.00	NULL
9	Erzuriz	12000.00	0.30
10	Ozer	11500.00	0.25
11	Cambraut	11000.00	0.30
12	Abel	11000.00	0.30
13	Raphaely	11000.00	NULL
14	Vishney	10500.00	0.25
15	Zlotkey	10500.00	0.20
16	King	10000.00	0.35
17	Tucker	10000.00	0.30
18	Bloom	10000.00	0.20
19	Baer	10000.00	NULL

10. El personal del departamento de Recursos Humanos desea tener mayor flexibilidad con los reportes hechos. Por ejemplo se requiere un reporte de los apellidos (last_name) y salarios (salary) de todos los empleados que tengan un salario mayor a un monto que el personal de Recursos Humanos ingresará. Probar con el valor \$ 12,000.

```
declare @salario as decimal(9,2); set @salario = 12000; select last_name,salary from employees where salary > @salario;
```

	last_name	salary
1	King	24000.00
2	Kochhar	17000.00
3	De Haan	17000.00
4	Greenberg	12008.00
5	Russell	14000.00
6	Partners	13500.00
7	Hartstein	13000.00
8	Higgins	12008.00

11. El departamento de Recursos Humanos requiere extraer reporte basados en el Administrador (manager_id). Se requiere crear una consulta que pregunte al usuario por el Administrador (manager_id) y genere un reporte con los números de empleado (employee_id), apellidos (last_name), salarios (salary) y numero de departamento de los empleados que este Administrador tiene a su cargo. Adicionalmente también se desea tener la habilidad de ordenar este reporte en base a una determinada columna. Probar con los siguientes valores:

Administrador (manager_id) = 103, ordenado por Apellido (last_name)

Administrador (manager_id) = 201, ordenado por Salario (salary)

Administrador (manager_id) = 124, ordenado por No de Empleado (employee_id)

```

declare @gerente as int;
set @gerente = 103;
select employee_id,last_name,salary,department_id from employees where manager_id = @gerente
order by last_name;
set @gerente = 201;
select employee_id,last_name,salary,department_id from employees where manager_id = @gerente
order by salary;
set @gerente = 124;
select employee_id,last_name,salary,department_id from employees where manager_id = @gerente
order by employee_id;
go

```

	employee_id	last_name	salary	department_id
1	105	Austin	4800.00	60
2	104	Ernst	6000.00	60
3	107	Lorentz	4200.00	60
4	106	Pataballa	4800.00	60

	employee_id	last_name	salary	department_id
1	202	Fay	6000.00	20

	employee_id	last_name	salary	department_id
1	141	Rajs	3500.00	50
2	142	Davies	3100.00	50
3	143	Matos	2600.00	50
4	144	Vargas	2500.00	50
5	196	Walsh	3100.00	50
6	197	Feeney	3000.00	50
7	198	OConnell	2600.00	50
8	199	Grant	2600.00	50

12. Generar un listado de apellidos (last_name) de todos los empleados que tengan la letra 'a' en la tercera letra de su apellido.

```
select last_name from employees where SUBSTRING(last_name,3,1) = 'a';  
go
```

	last_name
1	Grant
2	Grant
3	Whalen

13. Mostrar los apellidos (last_name) de todos los empleados que tengan tanto la letra 'a' como la letra 'e' en su apellido.

```
select last_name from employees where SUBSTRING(last_name,3,1) = 'a' or SUBSTRING(last_name,3,1)  
= 'e';  
go
```

	last_name
1	Greenberg
2	Chen
3	Gee
4	McEwen
5	Greene
6	Lee
7	Ozer
8	Abel
9	Grant
10	Fleaur
11	Everett
12	Feeney
13	Grant
14	Whalen
15	Baer
16	Gietz

14. Mostrar los apellidos (last_name), puestos (job_id) y salario (salary) de todos los empleados que sean Representantes de Ventas (SA_REP) o Responsables de Inventario (ST_CLERK) y cuyos salarios no sean iguales a \$ 2,500, \$ 3,500 o \$ 7,000.

```
select last_name,job_id,salary from employees where (job_id = 'SA_REP' or job_id = 'ST_CLERK')
and (salary = 2500 or salary = 3500 or salary = 7000);
go
```

	last_name	job_id	salary
1	Marlow	ST_CLERK	2500.00
2	Patel	ST_CLERK	2500.00
3	Rajs	ST_CLERK	3500.00
4	Vargas	ST_CLERK	2500.00
5	Tuvault	SA_REP	7000.00
6	Sewall	SA_REP	7000.00
7	Grant	SA_REP	7000.00

15. Modificar el reporte del ítem 4.6 y mostrar adicionalmente los datos de comisión (commission_pct) de todos los empleados que solamente el 20 % de comisión.

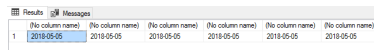
```
select last_name 'Empleado',salary 'Salario Mensual',commission_pct from employees where salary >5000 and salary <12000 and (department_id = 20 or department_id = 50) and commission_pct = 0.20;  
go
```

Empleado	Salario Mensual	commission_pct

5. Actividad No 05 – Funciones

1. Se requiere realizar una consulta que visualice la fecha del sistema.

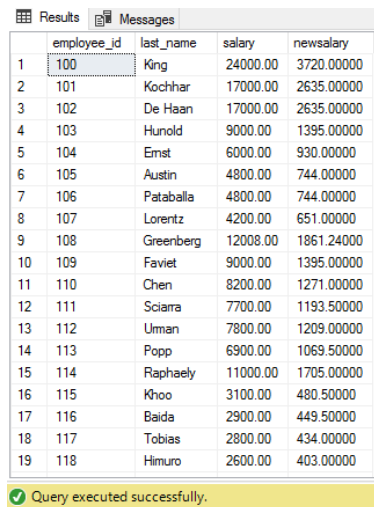
```
SELECT CONVERT (date, SYSDATETIME())
,CONVERT (date, SYSDATETIMEOFFSET())
,CONVERT (date, SYSUTCDATETIME())
,CONVERT (date, CURRENT_TIMESTAMP)
,CONVERT (date, GETDATE())
,CONVERT (date, GETUTCDATE());
```



	(No column name)	(No column name)	(No column name)	(No column name)	(No column name)
1	2019-05-05	2019-05-05	2019-05-05	2019-05-05	2019-05-05

2. El departamento de Recursos Humanos necesita un reporte de todos los empleados que muestre el No de Empleado, Apellidos, Salario y una columna más con el cálculo del salario incrementado en 15.5 % (expresado solo en enteros) esta columna debe etiquetarse Nuevo Salario

```
SELECT employee_id,last_name,salary,salary*0.155 as newsalary FROM employees
```



	employee_id	last_name	salary	newsalary
1	100	King	24000.00	3720.00000
2	101	Kochhar	17000.00	2635.00000
3	102	De Haan	17000.00	2635.00000
4	103	Hunold	9000.00	1395.00000
5	104	Ernst	6000.00	930.00000
6	105	Austin	4800.00	744.00000
7	106	Pataballa	4800.00	744.00000
8	107	Lorentz	4200.00	651.00000
9	108	Greenberg	12008.00	1861.24000
10	109	Faviet	9000.00	1395.00000
11	110	Chen	8200.00	1271.00000
12	111	Sciarra	7700.00	1193.50000
13	112	Uman	7800.00	1209.00000
14	113	Popp	6900.00	1069.50000
15	114	Raphaely	11000.00	1705.00000
16	115	Khoo	3100.00	480.50000
17	116	Baida	2900.00	449.50000
18	117	Tobias	2800.00	434.00000
19	118	Himuro	2600.00	403.00000

Query executed successfully.

3. Modificar la consulta anterior y adicionar una columna que muestre el resultado de la resta entre el antiguo salario y el nuevo salario. Etiquetar esta columna como Incremento.

```
SELECT employee_id,last_name,salary,salary*0.155 as newsalary,salary-(salary*0.155) as incremento FROM employees
```

Results		Messages			
	employee_id	last_name	salary	newsalary	incremento
1	100	King	24000.00	37200.00000	20280.00000
2	101	Kochhar	17000.00	26350.00000	14365.00000
3	102	De Haan	17000.00	26350.00000	14365.00000
4	103	Hunold	9000.00	13950.00000	7605.00000
5	104	Ernst	6000.00	9300.00000	5070.00000
6	105	Austin	4800.00	7440.00000	4056.00000
7	106	Pataballa	4800.00	7440.00000	4056.00000
8	107	Lorentz	4200.00	6510.00000	3549.00000
9	108	Greenberg	12008.00	1861.24000	10146.76000
10	109	Faviet	9000.00	1395.00000	7605.00000
11	110	Chen	8200.00	1271.00000	6929.00000
12	111	Sciarra	7700.00	1193.50000	6506.50000
13	112	Uman	7800.00	1209.00000	6591.00000
14	113	Popp	6900.00	1069.50000	5830.50000
15	114	Raphaely	11000.00	1705.00000	9295.00000
16	115	Khoo	3100.00	480.50000	2619.50000
17	116	Baida	2900.00	449.50000	2450.50000
18	117	Tobias	2800.00	434.00000	2366.00000
19	118	Himuro	2600.00	403.00000	2197.00000

Query executed successfully.

4. Crear un reporte que muestre los Apellidos (con la primera letra en Mayúsculas y las demás en Minúsculas) y la longitud de los apellidos (colocar alias Longitud), para todos aquellos empleados quienes sus apellidos empiecen con las letras 'J', 'A' y 'M'. Ordenar los resultados por la columna Apellido.

```
select UPPER(last_name) ^Apellido", (LOWER(first_name)) ^Longitud"
from employees
where last_name like 'A %'
or last_name like 'J %'
or last_name like 'M %' order by last_name asc;
```

Results		Messages	
	Apellido	Longitud	
1	ABEL	ellen	
2	ANDE	sundar	
3	ATKINSON	mozhe	
4	AUSTIN	david	
5	JOHNSON	charles	
6	JONES	vance	
7	MALLIN	jason	
8	MARKLE	steven	
9	MARLOW	james	
10	MARVINS	mattea	
11	MATOS	randall	
12	MAVRIS	susan	
13	MCCAIN	samuel	
14	MCEWEN	allan	
15	MIKKILINENI	irene	
16	MOURGOS	kevin	

Query executed successfully.

5. Modificar la consulta anterior a fin de que consulte primero al usuario con que letra empieza el apellido a buscar. Considerar que no importa si la letra esta mayúscula o minúscula de igual manera debe mostrar los resultados.

```
select initcap(FIRST_NAME) as "name", length(first_name) as "Length" from employees where upper(substr(first_name,1,1))=upper('&Inicial') order by first_name;
```

6. El departamento de Recursos Humanos la duración o tiempo de permanencia de cada empleado, mostrar el Apellido y el calculo del número de meses entre la fecha de hoy y la fecha en que fue contratado el empleado, Etiquetar la columna como Meses Trabajados, ordenar los resultados por el resultado de los números de meses, Redondear el número de meses al entero más cercano.

```
SELECT LAST_NAME, ROUND(MONTHS_BETWEEN(SYSDATE,HIRE_DATE),0) "MONTHS_WORKED"
from employees order by MONTHS_BETWEEN( HIRE_DATE, SYSDATE);
```

7. Crear una consulta que devuelva los Apellidos y Salarios de todos los empleados, Formatear la columna salario para que muestre 15 caracteres, completar con el símbolo '\$' los espacios previos al valor de la columna salario, ejemplo: \$\$\$\$\$\$\$\$\$\$10000. Etiquetar esta columna como Salario.

```
CREATE FUNCTION LPAD
(
  @string VARCHAR(MAX),
  @length INT,
  @pad CHAR
)
RETURNS VARCHAR(MAX)
AS
BEGIN
  RETURN REPLICATE(@pad, @length - LEN(@string)) + @string;
END
GO
SELECT dbo.LPAD(salary, 15, '$') VALUE
FROM employees;
```

Results		Messages
	VALUE	
1	\$\$\$\$\$\$\$24000.00	
2	\$\$\$\$\$\$\$17000.00	
3	\$\$\$\$\$\$\$17000.00	
4	\$\$\$\$\$\$\$9000.00	
5	\$\$\$\$\$\$\$6000.00	
6	\$\$\$\$\$\$\$4800.00	
7	\$\$\$\$\$\$\$4800.00	
8	\$\$\$\$\$\$\$4200.00	
9	\$\$\$\$\$\$\$12008.00	
10	\$\$\$\$\$\$\$9000.00	
11	\$\$\$\$\$\$\$8200.00	
12	\$\$\$\$\$\$\$7700.00	
13	\$\$\$\$\$\$\$7800.00	
14	\$\$\$\$\$\$\$6900.00	
15	\$\$\$\$\$\$\$11000.00	
16	\$\$\$\$\$\$\$3100.00	
17	\$\$\$\$\$\$\$2900.00	
18	\$\$\$\$\$\$\$2800.00	
19	\$\$\$\$\$\$\$2600.00	

✓ Query executed successfully.

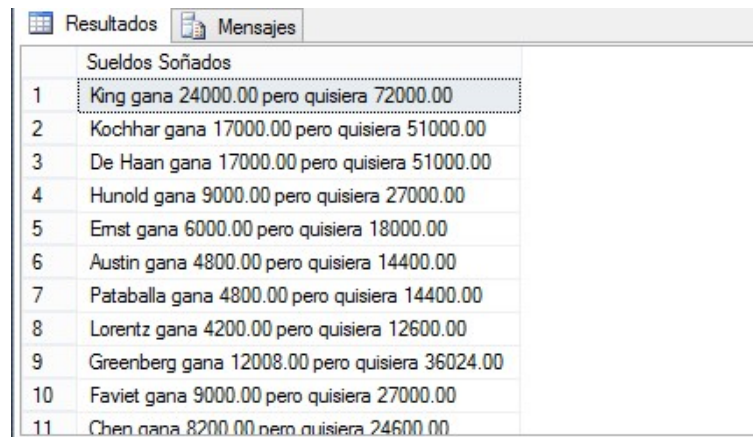
8. Crear una consulta que muestre en una única columna los primeros 8 caracteres del apellido de los empleados e indique sus salarios representados por asteriscos (*), cada asterisco representa el valor 1000. Ordenar el listado por el salario de los empleados. Asimismo Etiquetar la columna como 'Empleados y sus Salarios'.
9. Finalmente crear una consulta que muestre los Apellidos de los empleados y el No de Semanas Empleado hasta la actualidad para todos los empleados del departamento No 90, truncar el número de semanas a sin decimales. Ordenar el resultado por el No de Semanas y etiquetar la columna como tenencia.

```
select last_name, TRUNC(((SYSDATE-hire_date)/7),0) as TENURE from employees where department_id=90 ORDER BY hire_date DESC;
```

6. Actividad No 06 – Funciones de Conversión

1. Crear un reporte que muestre lo siguiente por cada empleado.
(Apellido del empleado) gana (Salario) pero quisiera (3 veces Salario).
Etiquetar la columna como Sueldos Soñados.

```
select 'Sueldos Soñados'=(last_name + ' gana ' + Cast(salary as varchar(18)) + ' pero  
quisiera ' + Cast((salary * 3) as varchar(18)))  
from dbo.employees  
go
```



	Sueldos Soñados
1	King gana 24000.00 pero quisiera 72000.00
2	Kochhar gana 17000.00 pero quisiera 51000.00
3	De Haan gana 17000.00 pero quisiera 51000.00
4	Hunold gana 9000.00 pero quisiera 27000.00
5	Ernst gana 6000.00 pero quisiera 18000.00
6	Austin gana 4800.00 pero quisiera 14400.00
7	Pataballa gana 4800.00 pero quisiera 14400.00
8	Lorentz gana 4200.00 pero quisiera 12600.00
9	Greenberg gana 12008.00 pero quisiera 36024.00
10	Faviet gana 9000.00 pero quisiera 27000.00
11	Chen gana 8200.00 pero quisiera 24600.00

2. Realizar una consulta que muestre el Apellido del empleado, fecha de contratación y la Fecha de Revisión del Salario, la cual es el primer Lunes después de cada seis meses de servicio, etiquetar la columna como Revisión, asimismo el formato de esta fecha debe ser similar al siguiente:

Lunes, el veintiuno de julio, 2003

```
select last_name, hire_date as Revision from employees  
where hire_date between '2003-06-17' and '2005-09-21';  
go
```

Resultados		Mensajes
	last_name	Revision
1	King	2003-06-17
2	Kochhar	2005-09-21
3	Austin	2005-06-25
4	Tobias	2005-07-24
5	Weiss	2004-07-18
6	Fripp	2005-04-10
7	Nayer	2005-07-16
8	Bissot	2005-08-20
9	Marlow	2005-02-16
10	Mallin	2004-06-14
11	Ladwin	2003-07-14

3. Mostrar un reporte que tenga los Apellidos, Fecha de Contratación y el Día de Inicio de cada empleado (Lunes, Martes, etc...), etiquetar la última columna como Día. Ordenar los resultados por el Día de Inicio empezando por Lunes.

```
select e.last_name, e.hire_date, DateName(WEEKDAY, jh.START_DATE)as 'Dia'
from dbo.employees as e inner join dbo.job_history as jh on
e.employee_id=jh.employee_id
go
```

Resultados		Mensajes	
	last_name	hire_date	Dia
1	Kochhar	2005-09-21	Domingo
2	Kochhar	2005-09-21	Domingo
3	De Haan	2001-01-13	Sábado
4	Raphaely	2002-12-07	Viernes
5	Kaufling	2003-05-01	Lunes
6	Taylor	2006-03-24	Viernes
7	Taylor	2006-03-24	Lunes
8	Whalen	2003-09-17	Domingo
9	Whalen	2003-09-17	Lunes
10	Hartstein	2004-02-17	Martes

4. Crear un listado que muestre los Apellidos de los empleados y sus Montos de Comisión, en caso no tenga comisión deberá mostrar el texto 'Sin Comisión', etiquetar esta ultima columna como Comisión.

```

select last_name as 'Apellidos', 'Comision'='Sin Comision' from dbo.employees where
commission_pct = 0
UNION
select last_name as 'Apellidos', 'Comision'= Cast((salary * commission_pct) as
varchar(20)) from dbo.employees where commission_pct >0
go

```

Resultados		Mensajes
	Apellidos	Comision
1	Abel	3300.0000
2	Ande	640.0000
3	Banda	620.0000
4	Bates	1095.0000
5	Bernstein	2375.0000
6	Bloom	2000.0000
7	Cambrault	1500.0000
8	Cambrault	3300.0000
9	Doran	2250.0000
10	Errazuriz	3600.0000
11	Fox	1920.0000

5. Utilizando la función DECODE, crear un reporte que muestre los apellidos, los puestos y los grados de los empleados basados en sus puestos, utilizando la siguiente información:

Puesto	Grado
AD_PRES	A
ST_MAN	B
IT_PROG	C
SA_REP	D
ST_CLERK	E
Ninguno de los Anteriores	0

6. Rescribir la consulta anterior utilizando la función CASE.

```

select e.last_name as 'Apellidos', j.job_title, case
when j.job_id = 'AD_PRES' THEN 'A'
when j.job_id = 'ST_MAN' THEN 'B'
when j.job_id = 'IT_PROG' THEN 'C'
when j.job_id = 'SA_REP' THEN 'D'
else '0' END as 'Grados' from dbo.employees as e inner join dbo.jobs as j on
e.job_id=j.job_id
go

```


Resultados		Mensajes	
	Apellidos	job_title	Grados
1	King	President	A
2	Kochhar	Administration Vice President	0
3	De Haan	Administration Vice President	0
4	Hunold	Programmer	C
5	Ernst	Programmer	C
6	Austin	Programmer	C
7	Pataballa	Programmer	C
8	Lorentz	Programmer	C
9	Greenberg	Finance Manager	0
10	Faviet	Accountant	0
11	Chen	Accountant	0